**Observation 39-enforcing complex data integrity values**

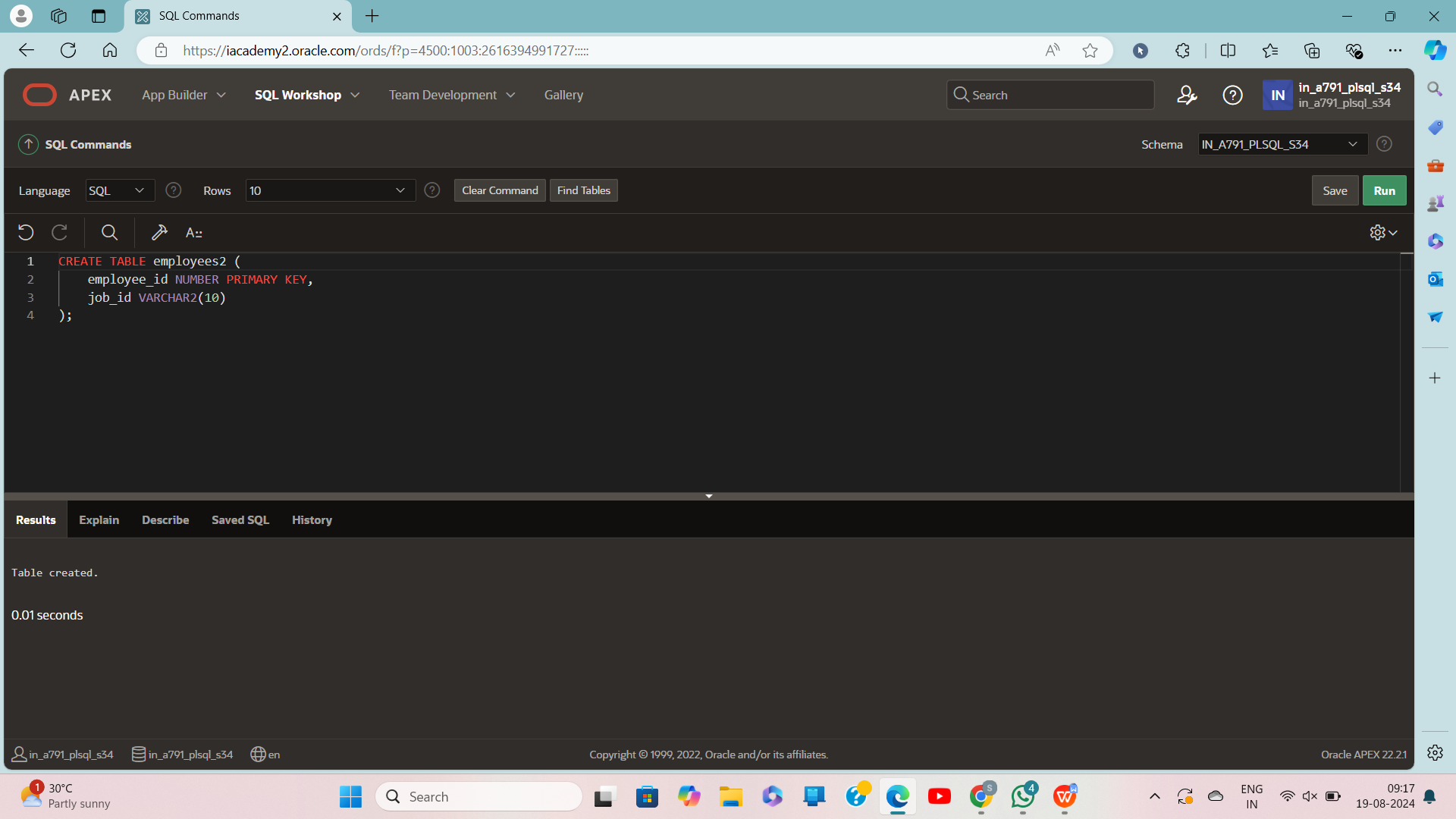
Create table:

CREATE TABLE employees2 (

employee\_id NUMBER PRIMARY KEY,

job\_id VARCHAR2(10)

);



Create job:

CREATE TABLE job\_data (

employee\_id NUMBER,

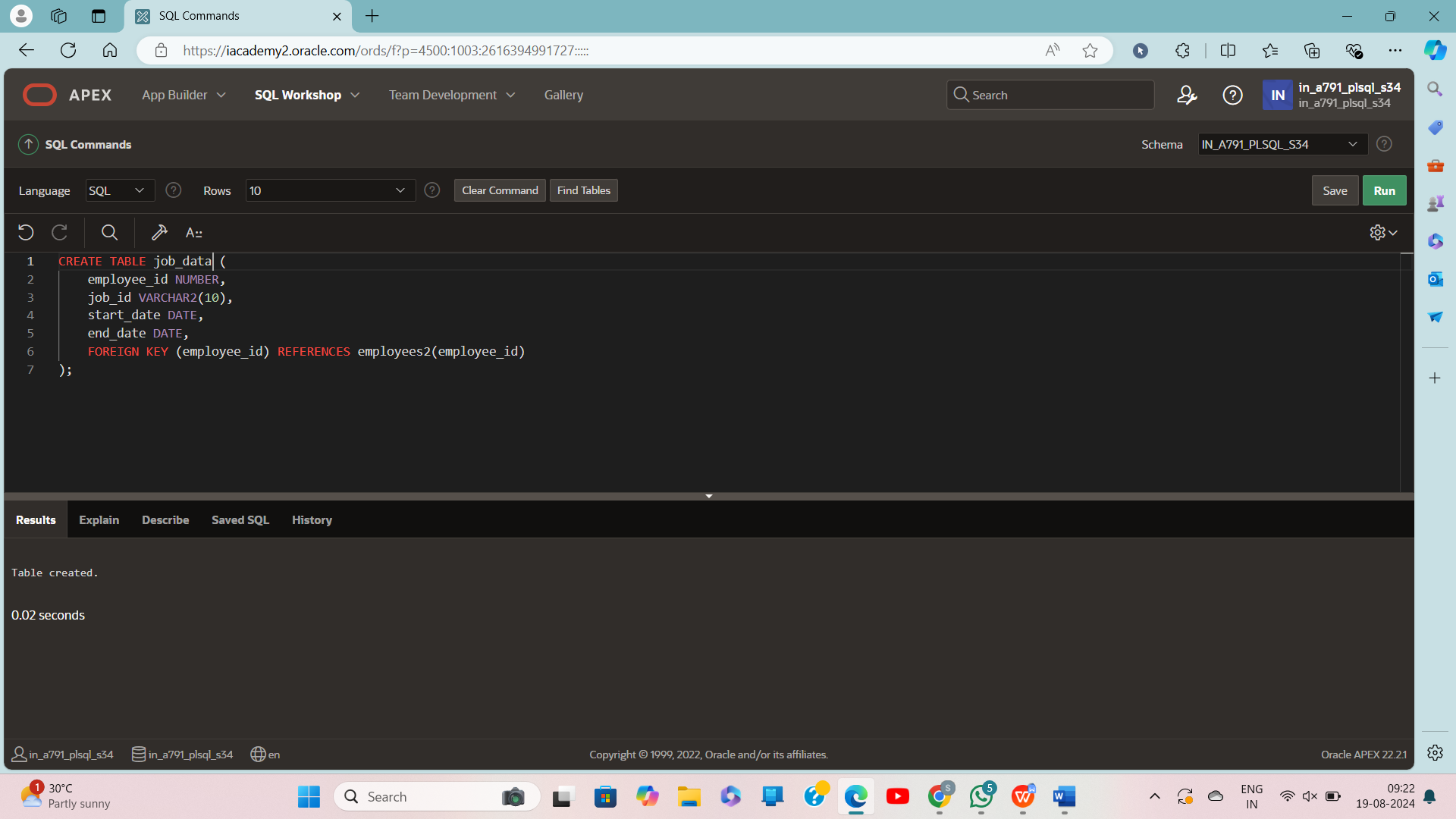
job\_id VARCHAR2(10),

start\_date DATE,

end\_date DATE,

FOREIGN KEY (employee\_id) REFERENCES employees2(employee\_id)

);

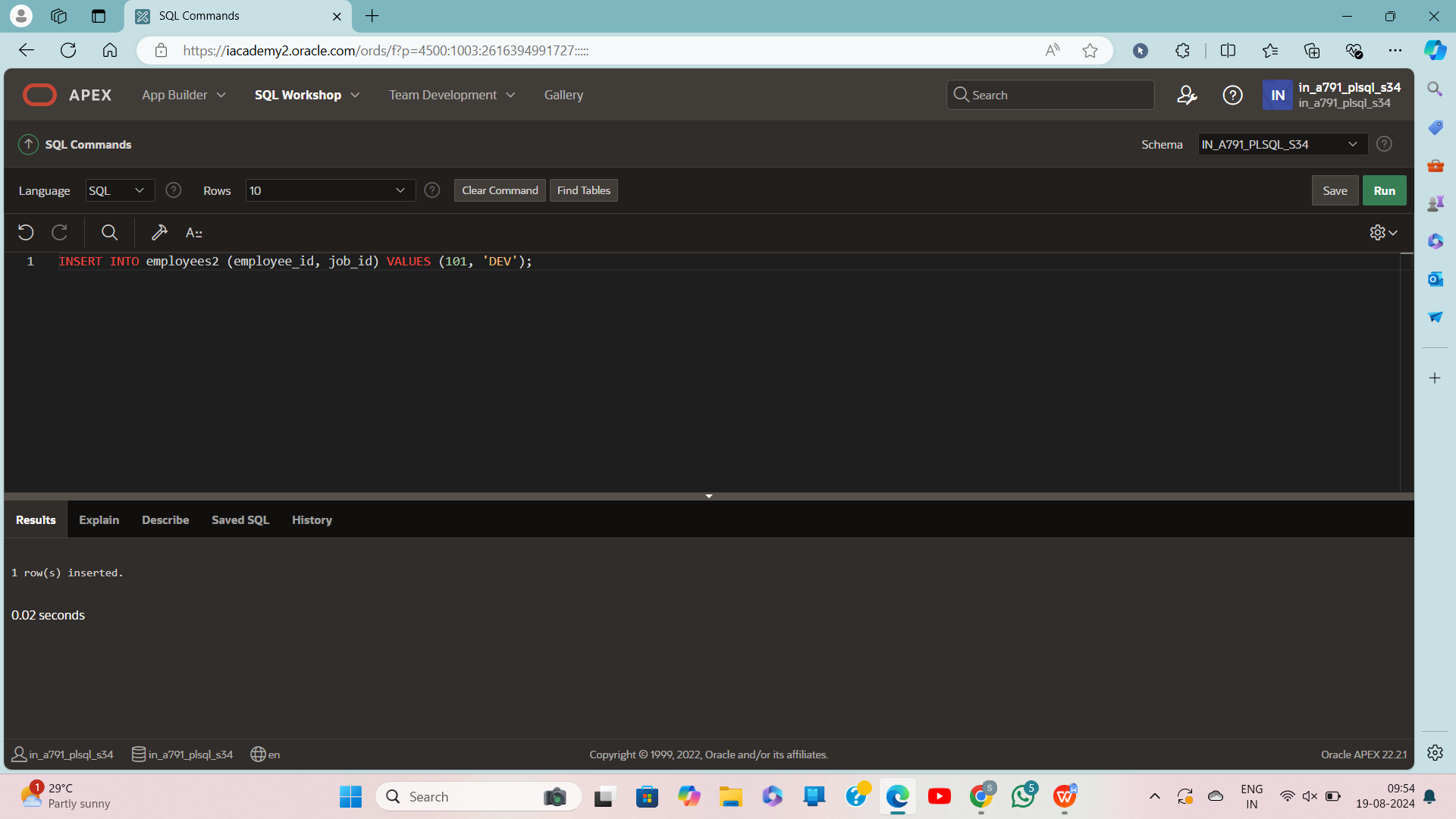


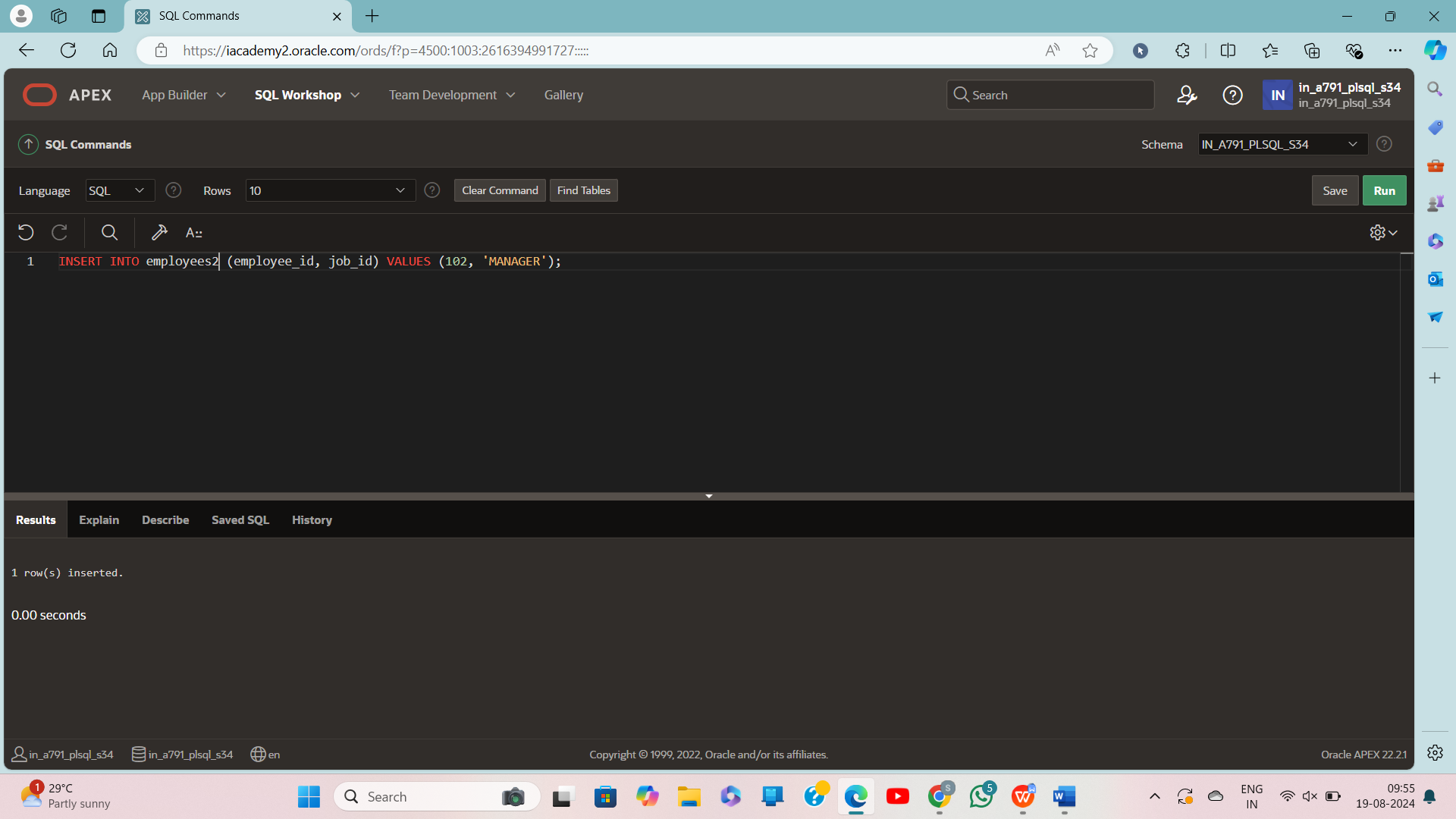
**Insert values:**

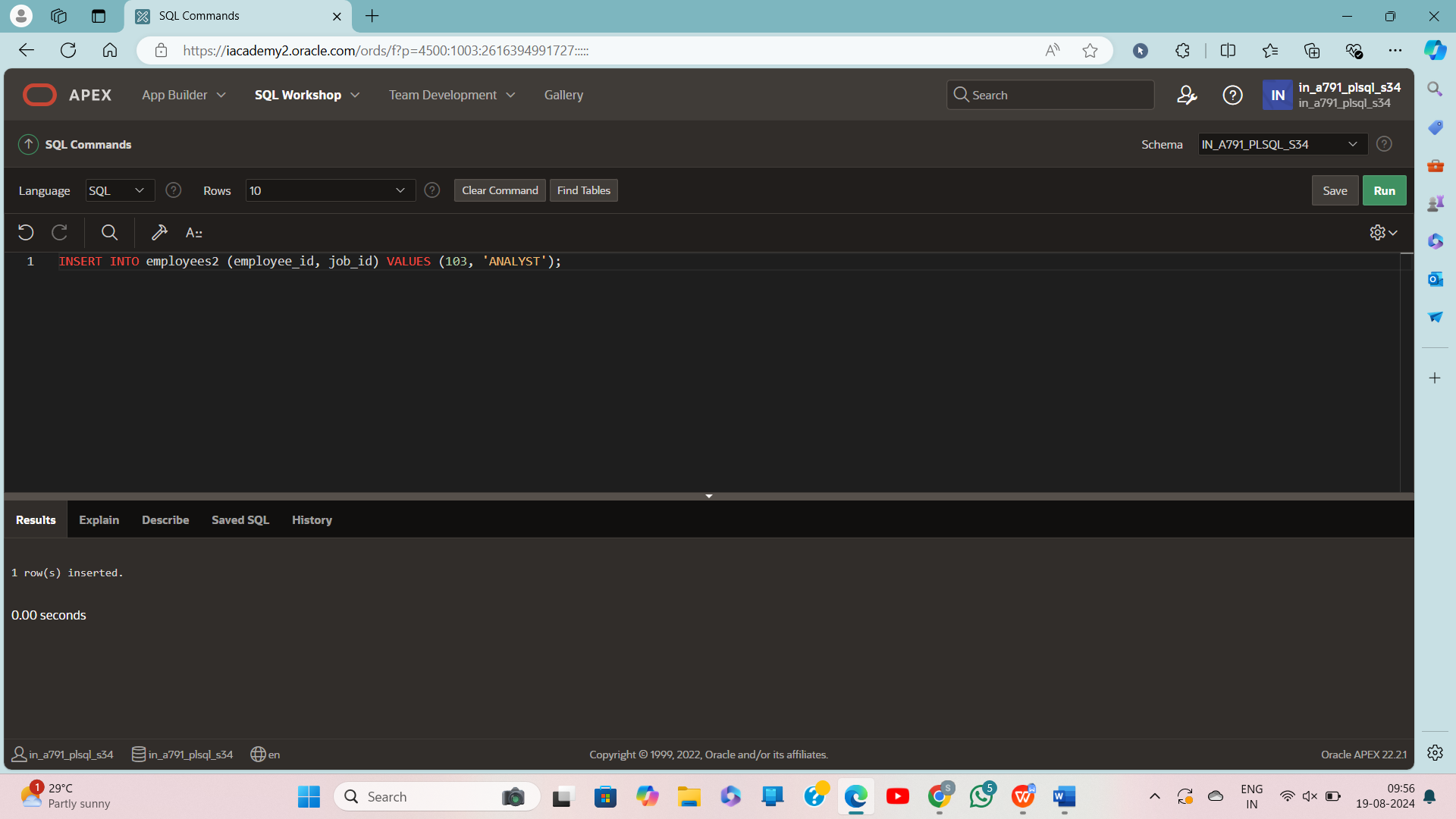
INSERT INTO employees2 (employee\_id, job\_id) VALUES (101, 'DEV');

INSERT INTO employees2 (employee\_id, job\_id) VALUES (102, 'MANAGER');

INSERT INTO employees2 (employee\_id, job\_id) VALUES (103, 'ANALYST');





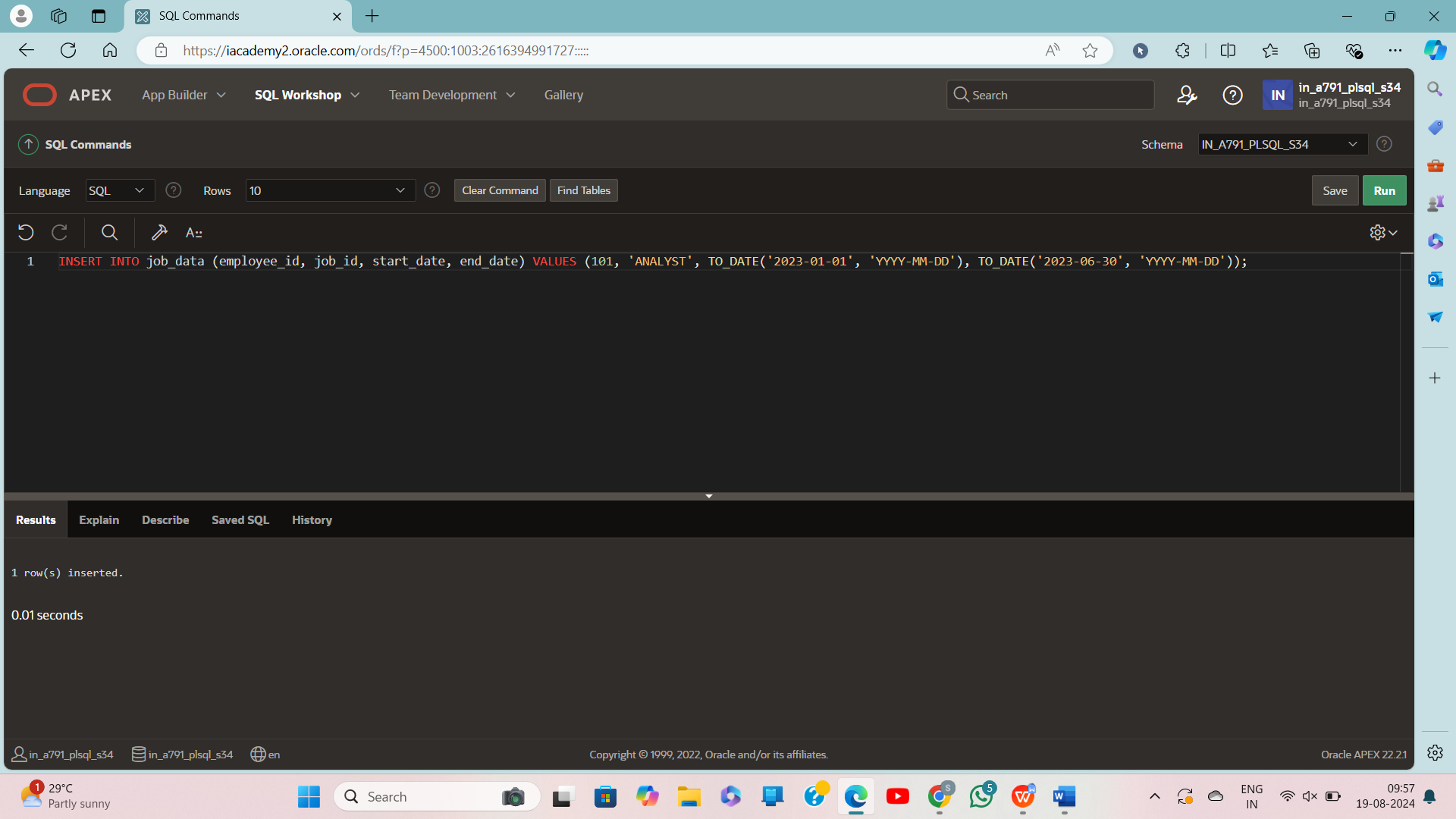


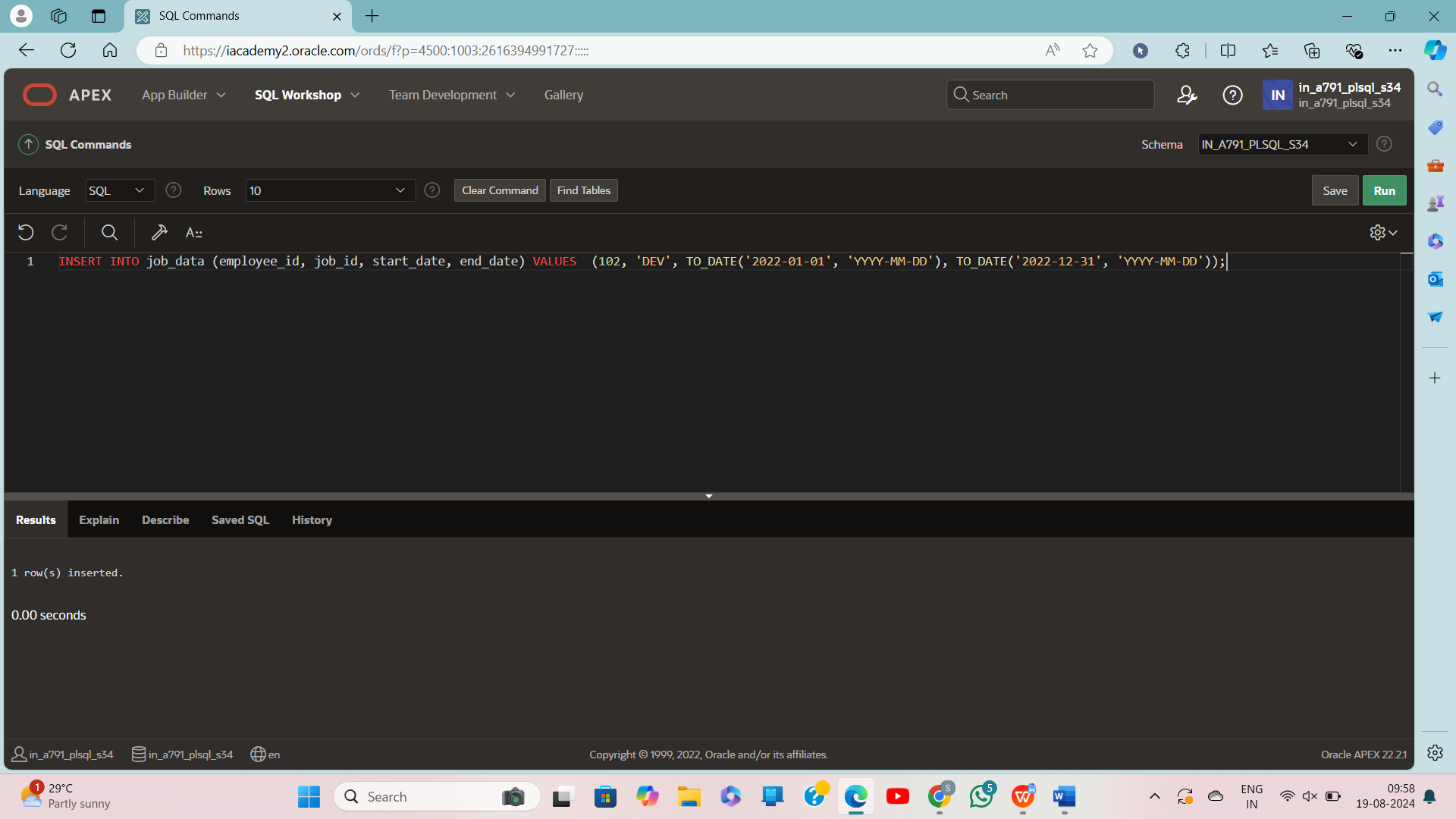
Insert in job:

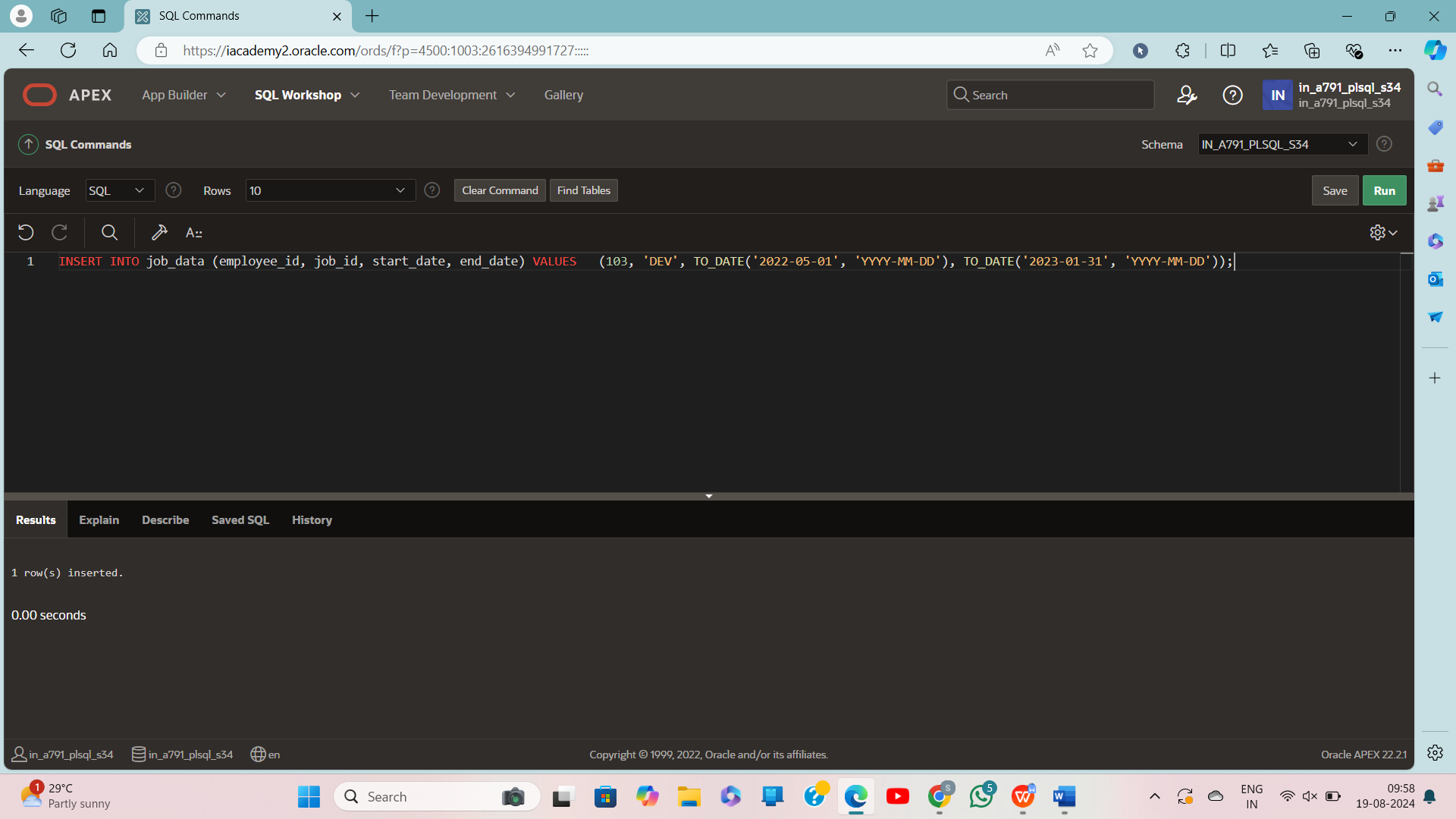
INSERT INTO job\_data (employee\_id, job\_id, start\_date, end\_date) VALUES (101, 'ANALYST', TO\_DATE('2023-01-01', 'YYYY-MM-DD'), TO\_DATE('2023-06-30', 'YYYY-MM-DD'));

INSERT INTO job\_data (employee\_id, job\_id, start\_date, end\_date) VALUES (102, 'DEV', TO\_DATE('2022-01-01', 'YYYY-MM-DD'), TO\_DATE('2022-12-31', 'YYYY-MM-DD'));

INSERT INTO job\_data (employee\_id, job\_id, start\_date, end\_date) VALUES (103, 'DEV', TO\_DATE('2022-05-01', 'YYYY-MM-DD'), TO\_DATE('2023-01-31', 'YYYY-MM-DD'));







**Create trigger:**

CREATE OR REPLACE TRIGGER check\_sal\_trigg

BEFORE UPDATE OF job\_id ON employees

FOR EACH ROW

DECLARE

v\_job\_count INTEGER;

BEGIN

-- Check if the new job\_id already exists in the job\_history for the same employee

SELECT COUNT(\*) INTO v\_job\_count

FROM job\_history

WHERE employee\_id = :OLD.employee\_id

AND job\_id = :NEW.job\_id;

-- If the job has already been done, raise an error

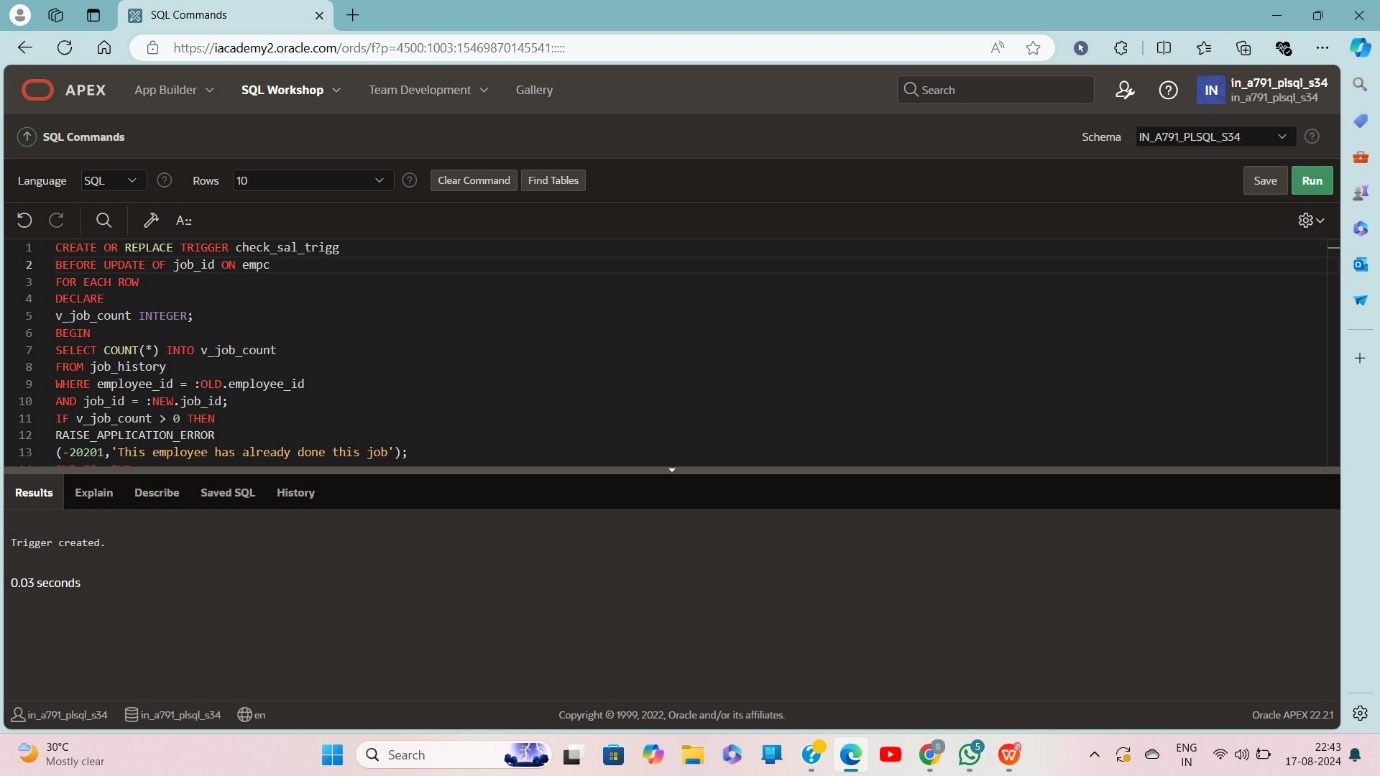
IF v\_job\_count > 0 THEN

RAISE\_APPLICATION\_ERROR(-20201, 'This employee has already done this job');

END IF;

END;

/



select\*from job\_data

